

DIAGNOSIS
BASED ON INCOMPLETE CAUSAL MODEL

*A Thesis Submitted
in Partial Fulfilment of the Requirement
for the Degree of*

MASTER OF TECHNOLOGY

by

JYOTIBIKAS BHATTACHARYA

to the

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY
KANPUR

July 1991

CERTIFICATE

This is to certify that the thesis work entitled **Diagnosis Based On Incomplete Causal Model** has been carried out by Jyotibikas Bhattacharya under my supervision and has not been submitted elsewhere for a degree.



(Dr. R. M. K. Saha)

July 1991

Kanpur

Professor

Dept. of Computer Science

and Engineering

Indian Institute of Technology

Kanpur

C-L-191 M
F111 D

CSC-1991-M- BHA-DIA

9 2 NOV 1991

CENTRAL
II A
1412228
ARY

Acc. No A

ACKNOWLEDGEMENTS

I am extremely grateful to Dr R M K Sinha for his valuable guidance without which it would have been difficult to complete this work

I express my sincere gratitude to MOTOROLA Inc for providing financial support to the last stages of my thesis work

I would also like to thank all other colleagues and staff members of the department for providing me with an excellent working environment

Thanks are also due to the doctors and other staff members of the local Health Centre for attending to my medical exigencies. Special thanks are due to Dr A Sinha for patiently explaining the many concepts in my maiden intrusion into the medical domain

Special thanks are also due to all those people who have made my stay at IIT Kanpur a memorable one. I also thank my parents and other family members for their emotional support and understanding extended continuously to me

July 1991

Jyotibikas Bhattacharya

ABSTRACT

In the present work a mechanism for diagnosis of multiple faults in a system has been implemented. The domain knowledge is represented at the deep level by a causal network and at a shallow level by rules. Two forms of incompleteness are accommodated in the model. First, the initial causes or perturbations responsible for the malfunctioning of the system under consideration are assumed to be nonverifiable. Secondly, the relations CAUSES and HAM in the causal network may be incomplete and can be labelled a MAY denoting such incompleteness. This second form of incompleteness implies that certain conditions are being abstracted. The diagnostic process proceeds in iterations of set covers over the set of all observations. The manifestations not observed are assumed to be absent. The diagnostic process is non monotonic. It can also lead to multiple solutions. The criterion of parsimony over initial causes has been used to generate a solution which ensures the assumption of minimum number of initial cause. Such a solution has been termed an IC Parsimonious solution. After having succeeded in generating such a solution based on reasoning in the deep level, the system abstracts shallow knowledge from this in the form of rules. When a diagnostic problem is posed, the system tries to generate a solution using the shallow knowledge. If it does not succeed, then it uses the deep knowledge to generate a solution.

CONTENTS

	Page
Chapter 1 Introduction	0
1.1 Organization of the Thesis	0
Chapter 2 A Review of Diagnostic Systems	0
2.1 What is Involved in Building a Diagnostic System	0
2.2 Deep versus Shallow Knowledge	1
2.3 Diagnostic Systems Some Examples	1
2.3.1 The MYCIN System	1
2.3.2 The MIND System	1
2.3.3 The GDE System	1
2.3.4 The CASNET System	1
2.3.5 The Hanna & Gold System	1
2.4 Where the Present Work Fits in	2
Chapter 3 System Level Overview	2
3.1 A Formalism to Represent Deep Causal knowledge	2
3.2 Diagnostic Problems and Their Solutions	2
3.3 Parsimonious Solution	3
3.4 Problem Formulation & Generation of	
IC Parsimonious Solutions	3
3.5 An Example	3

Chapter 4 Implementat on Details

4 1 Data Structures

4 2 Implementat n of the Various Module

4 2 1 Module fo mN t

4 2 Module formDef

4 2 3 Module c mpile

4 2 4 Modul g tOb

4 5 Module solve

Chapter 5 Conclusions

5 1 Conclusions

5 2 Imp ov m nts and Scope for Future Work

REFERENCES

APPENDIX A Sample Network Formation

APPENDIX B Sample Diagnostic Session

LIST OF FIGURES

	Page No
Examp1 Rul s in MYCIN	14
An Example Causal Network	25
Schemati Structure of the Diagn tic System	34

Chapter 1

INTRODUCTION

Diagnosis is known to mankind from the dawn of civilization. In the beginning it started with medical diagnosis but later on as man developed different artifacts he also had to diagnose their faults when they needed repair. It so turns out that the human physiological system because of its sheer complexity has remained the most enigmatic of all systems that require fault diagnosis. It is perhaps because man has himself built the electronic and mechanical systems that he understands the working of these systems better than his own physiological system and hence can diagnose faults more precisely in case of the former. This indicates that some deep knowledge about the system helps one in diagnosing its faults. But then how often do we use or more relevantly in a position to have and hence use such deep knowledge about systems? Indeed a faulty television is faster repaired by a TV mechanic rather than many an electronics engineer who has a deeper knowledge about the system. A mechanic's knowledge about the system is indeed superficial but his experience in diagnosis and the tricks of the trade learnt from his master equip him with lots of heuristic knowledge about associating manifestations of faults with subsystems that are faulty. He does not have the lower level causal functional and physical information of the system but such knowledge is not needed in the more common

ca s However when a new problem appears the
m h a i c s r e p e t o r of knowledge fails to meet the situation
That is when we need the deep knowledge of the engineer to
fall back upon. Somewhat similar situ tio prevails even in th
case of medical diagnosis. The doctor s exp rien e equips him
with a lot of rule about correlating diseases with their
manife tat ons (ymptom of th disea es). First he tries to
apply such knowledge to arrive at a diagnosis. Only when he
fails he falls back on his deep knowledge to carry out a
d tailed diagnos. That is the expert can reason from first
principles if necessary to find a solution. Thus in domains such
as medical diagnosis deep knowledge must be modelled.

Diagnostic problems have often been cast into a pattern
recognition or statistical d ci on th ory framework. Computer
repr sentation is not difficult and as a result many well-known
methods such as those based on Baye Theo em have been used. The
difficulties with applying these methods (e g scarcity of
statistics and the use of invalid approximations) are also
sufficien ly persistent so that alternative approaches have been
sought. In many medical areas existing knowledge could nhance
the decision making capabilities of a diagno tic system. There
are many useful decision rules specifi to a given medical
application that the physician directly applies in his or her
reasor ng

The statistical methods extract the the decision making rules implicitly from accumulated sample experience. The approaches based on Artificial Intelligence (henceforth abbreviated as AI) attempts to capture such rules more explicitly and intend to overcome some of the limitations of purely statistical methods by developing a more structured representation for the diagnostic and the appropriate selection problems. A program that uses decision strategies based on explicit representations of medical knowledge can more easily incorporate evolving changes in its knowledge base independent of the reasoning strategies. It can also incorporate the results of clinical experience by matching the more explicit patterns of reasoning to the decisions and opinions of physician. Such systems are more likely to be accepted because they are expressed in a decision making context familiar to the clinician. A structured representation can also permit the formulation of complex hypotheses that express progression and severity of disease. Some researchers have attempted to increase the scope, accuracy and explanation capabilities of their systems by increasing structure while still preferring a statistical framework. Others have relied on logical and semantic encodings of contextual knowledge within an artificial intelligence framework.

Several fundamental AI issues are raised by medical decision making problems. One important issue concerns the development of representations that are powerful enough to capture a complex and

hanging knowledge base in a realistic task domain Method of acquiring knowledge from experts the choice of appropriate levels of abstraction and resolution for describing a given problem and the choice of a computer representation of the knowledge base are all problems that immediately arise in developing such systems They are closely linked to the control strategies or methods used to produce interpretations for individual cases Fundamental to most such control strategies is the capability of approximate reasoning This is needed to manage the multiple hypotheses that can be generated from a large and complex knowledge base which includes statements at different levels of uncertainty Once decisions are reached producing explanations becomes an important task if the acceptability of the system is enhanced Practical issues of implementation for these large knowledge based systems include ease of knowledge management (updating) efficiency choice of languages and transferability into practical use in both the original domain and other similar ones

The rapid evolution in digital technology has made problems related to testing digital systems particularly difficult to solve Some of these problems deal with test generation fault diagnosis design for testability and checking for the compliance of design for test rules Over the last two decades numerous systems have been developed to solve these problems However in order to develop more powerful and possibly more efficient

systems researchers have looked toward the use of AI and specifically knowledge based technology to provide new approaches. Fault diagnosis deals with localization of faults or defects based on observed system behaviour. The diagnostic process requires knowledge about the hardware under diagnosis. Different approaches require different types of knowledge. In an extreme case if the system can be modelled as a chain of functionally dependent components minimum knowledge is required for diagnosis. By probing the output and input of each component in the chain a troubleshooter can determine which component is faulty. For example if the input to a component is determined to be correct and the output is observed to be incorrect then this component is assumed to be faulty. The only knowledge required to diagnose such a system is the dependency chain. Reasoning is simple in this case. However most systems do not have such a chain structure and thus require a more complex diagnostic procedure. For these cases much more knowledge about the system is required and AI techniques are often employed in the diagnostic process.

The difficulties encountered in applying knowledge based systems technology to complex systems have made the need for representing and using deep knowledge about system behaviour increasingly clear to designers. This knowledge is typically well structured formal relying on established theories. As opposed to empirical

additional knowledge such knowledge is said to provide a deep model of the domain. There has also been attempts to form a general theory of diagnosis from first principles based on deep models. But such models rarely incorporate any form of incompleteness. The present work aims at identifying and representing the causal structure underlying a specific expertise with some forms of incompleteness incorporated in the representation formalism. Some shallow knowledge in the form of rules is abstracted after reasoning in the deep model and finding out a parsimonious solution.

1.1 Organization of the Thesis

In chapter 2 a review of the related work is presented. Different approaches are discussed and examples are briefly described.

In chapter 3 the theoretical basis of the implemented system followed by a system level description of the implementation is presented.

In chapter 4 the implementation details of the various modules present in the diagnostic system are presented.

In chapter 5 conclusions and direction toward future work are indicated.

Chapter 2

A REVIEW OF DIAGNOSTIC SYSTEMS

We present here a brief review of several diagnostic systems that have been developed over the years. But before that let us have a look at what is in general involved in building a diagnostic system.

2.1 What is Involved in Building a Diagnostic System ?

Before building a diagnostic system one should address the following question within the scope of a given application: how can we define an observation? what do we call a symptom? What do we call a failure and how do we express it?

A conceptual analysis of the diagnostic process undertaken from concrete applications has led researchers to propose a triple problem breakdown viz observation interpretation qualification.

It is obvious that any diagnosis depends on the observation of a real object. Therefore description capacities of what is observed are needed either implicitly or explicitly. All diagnostic systems include an implicit or explicit description language that defines and limits the information used by the system to carry the diagnosis.

The choice of this language is fundamental. It can be very abstract (value graphs, timing diagram etc.) and its adequacy will condition the final performance. In fact the system will try to make a diagnosis on the mere basis of the phrases in the language. The latter should be sufficiently complete to enable the easiest diagnosis possible but not too complex so as to simplify the observation process (indeed it is much easier to measure voltage than to establish a timing diagram).

The data observation phase brings in the notion of correct frame of reference (or description of nominal behaviour). The observed object must be compared to a nominal model.

For an electrical appliance the reference could be a complete state graph. In the case of an electronic board the correct function model may be an aggregate of equations that checks the inputs and outputs of each elementary component. By simulation one could then obtain the nominal state of the board given a test vector. The comparison between the theoretical and measured values would then give the fault indexes.

For more complex and non decomposable systems a symptomatological model can be used. A patient may be said to be feverish by knowing that the normal state is the absence of fever, a car engine that chatters by knowing that the normal state is a steady running rate. The very term of symptom thus makes a reference to the observation of an abnormal state. If the description capacities of an object are limited to symptoms then

(implicit) reference of correct functioning is defined by the absence of any symptoms. This is a form of negative reference but it is not really different from a more constructive model like the equational model.

Now to perform a diagnosis one bases it on the state of the object and the explicit or implicit comparison to a reference. The problem then remains the problem of the fault description itself that we have called fault qualification. Let us suppose that we have a number of observations and a reference at our disposal and that we are in a position to check if that reference is respected or not. If it is not, we could limit ourselves by showing the incompatible measures and equations and to then state that the fault is described by these observations and this reference. The execution of a diagnostic process as it is intuitively perceived must go further.

For example with an electronic board we do not want to show a description of divergences but to put a name to the component that is supposedly responsible for these divergences. What we mean here is failure qualification by component identification. We therefore find ourselves a priori a supplementary constraint which is to merely describe the failure in terms of components.

In the same way in medicine an illness could be defined as a list of symptoms since each symptom is an anomaly in itself. The qualification of failures in this case is by illness identification.

Diagnostic problem description therefore passes by the specification

of perceptive capacities

of a correct function reference

of a failure qualification language

The diagnosis process environment is then perfectly defined. The choice of the format of these entities is however totally subjective.

Now an algorithm is to be written which manipulates the data and makes the fault hypotheses based on a nominal model of the observed object correspond to the observations.

Algorithmic problems linked to this final stage may entail a rethinking of the formats (sophistication to reduce the amount of work required or thinning down to simplify the algorithms).

In general the approach is mixed in that it is often known beforehand what type of algorithm will be arrived at (graph search, pattern matching, etc) and that this information is used to guide the choice of format.

An efficient strategy for fault diagnosis relies upon specific symptoms to activate only a small portion of the available diagnostic knowledge. However, expert systems and humans using this strategy often commit characteristic errors. The errors commonly referred to as garden path errors occur because significantly different faults may manifest similar symptoms. If these symptoms are confused, the wrong portion of diagnostic knowledge can be activated. The Galen Expert System [Moen88] investigates occurrence of such errors and describes a strategy for reducing their occurrence.

2.2 Deep versus Shallow Knowledge

Two major types of knowledge are commonly used in diagnostic reasoning: namely, shallow knowledge and deep knowledge.

Shallow knowledge refers to the empirical knowledge of human experts in the association of the observed symptoms, which is the discrepancies between the predicted and observed behaviour with system faults. This knowledge is usually based upon prior machine deductions, statistical intuition or the past experience of human experts. It is primarily heuristic and is limited to a specific domain. Such knowledge often results in impressive performance when it is of the form which can be put into a rule format. However, outside the scope of expertise of the knowledge source, the performance is poor. Also, shallow knowledge often does not take into account the structure or design of the system being

diagnosed. Examples of such systems are MIND (developed by A J Wilkinson et al) MYCIN (developed by E Shortliffe et al) ATEX (developed by M B Bassant et al) etc

Deep knowledge refers to knowledge about the structure and behaviour of the system under test in addition to knowledge of diagnostic methodologies. Reasoning directly from deep knowledge is often referred to as first principles or model based reasoning. Descriptions of the system structure and behaviour are used which can exist in multiple levels of abstraction. The knowledge is deep in the sense that it is possible to explain the misbehaviour down to very low level components of the system as long as that level of description is available. Examples of such systems are GDE (developed by J De Kleer et al) CASNET ([Weiss78]) [Reiter87] etc

Some diagnostic systems use both deep and shallow knowledge. The shallow knowledge is first used to relate symptoms to possible causes of the malfunction and supply appropriate testing information. If the process fails the system then uses deep knowledge. The combined knowledge can be used to diagnose system efficiently. Examples of such systems are [Hanna88] IDM [Fink87] SIDI [Ayeb88] etc

2.3 Diagnostic Systems Some Examples

We take up some example systems each of which uses one of the approaches described in the previous section.

2 3 1 The MYCIN System

MYCIN [Clancey84] is an interactive program designed to be used as a consultant in difficult cases of meningitis or bacteraemia. It suggests a set of likely organisms (bacteria) and their appropriate therapy that will treat those that are most significant. MYCIN was one of the first rule-based expert systems. The domain knowledge of MYCIN is represented in a set of about 500 production rules. Most of the rules encode associations between findings and a hypothesis. Problem-solving behaviour could be modified by altering a rule or adding new ones. However, valuable knowledge about disease taxonomy, cause and effect, and temporal ordering between disorders are represented only implicitly, somewhat buried in rules. This frustrated attempt to use MYCIN's rules for intelligent, computer-aided instruction. MYCIN backward chains through the rule base, although for reasons of efficiency, rules are at times invoked in a data-directed fashion. Rules were meant to represent only domain knowledge, but ultimately they do need a good deal of control logic as well. Meta-rules are used for controlling the invocation of other rules. Meta-rules are associated with clinical parameters; they are strategy rules which suggest the best approach to establish the value/s of the associated parameter. Figure 1 contains some sample rules. Later rule-based systems have attempted to achieve a cleaner

Object rule with a priori certainty factors shown in brackets

```
IF      1) the site of the culture is the same AND
      2) the identity of the organism is the same
THEN    there is suggestive evidence (0.9) that the subtype of
      the organism is in group D
```

Meta rule for the identity parameter of organism states

```
IF      1) the site of the culture is one of the notable
      sites AND
      2) there are rules which mention in their premise a
      previous organism which may be the same as the current
      organism
THEN    it is definite (1.0) that each of them is not going to be
      useful
```

Figure 1

Example rules from the MYCIN system

epa ation b t w en cont ol a d domain le el knowledg The MYCIN research is well known for its production ule representation a d als for it i no at i e model of inexact reasoning the calculus of c tai ty fa tors MYCIN became a laboratory for in stigations into knowledg acquisition metalevel reasoning intelligent computer aid d nstruct on xplanatio a d kn wl dg e gine ing to l

2 3 2 The MIND System

The MIND ystem [Wilkinson85] is a rule ba d exp t system design d for VLSI diagno is It uses shallow knowledg and integrates the principles of hierarchical design and an expert s he ristics to achieve an efficient diagnostic procedure The main elements n the MIND system include th following the Check program test system hardware with built in self test (BIST) features a test knowledge database a diagnostic rule database causal maps an inference engine and a user interface The Check program ensures that the test system is wo king at a functional level It provides stimulus fo the BIST hardwa of the test system and collects test results(symptoms) which may indicate the misbehaviour of a certain function without knowing where fa lt is located These symptoms are then stor d in the test knowledge database for lat r use by a repair expert inferenc engine

A major feature of this system is its speed in processing rules. All rules are converted into integer or a tokenized form and then input to the inference engine. This leads to a much faster evaluation of rules.

One advantage of this system is that it integrates the expert diagnostic technique with the automatic symptoms detection capability. Using the Check program, the MIND system can detect symptoms automatically. The process of a machine interrogating a user for symptoms, which is prone to errors, thus can be avoided.

2.3.3 The GDE System

The General Diagnostic Engine (GDE) [DeKleer87] is a diagnostic system that exploits deep knowledge. But unlike many other systems, the space of potential candidates when encountering the possibility of multiple faults does not grow exponentially with the number of faults considered. To diagnose multiple faults, the features of ATMS (Assumption Based Truth Maintenance System) are used.

Measurements are needed to actually determine if a candidate component is faulty. In determining the next measurement, which leads to the discovery of the faulty components, using a minimum number of measurements. GDE uses an ATMS. It incorporates the a priori probabilities of individual component failure such that it can easily compute the conditional probability of a candidate.

and the possible outcome of measurements based on a model of the potentially faulty device

This combination of probabilistic inference and ATMS enables GDE to apply a minimum entropy method to determine what measurement to make next. The best measurement is the one which minimizes the expected entropy of candidate probabilities resulting from the measurement.

2.3.4 CASNET

The CASNET [Weiss78] system is in principle a general tool for building expert systems for the diagnosis and treatment of diseases whose mechanisms are well known. It uses deep knowledge. Diseases are modelled in terms of Causal Association Networks. Such a network is a particular type of semantic network designed to

- describe dynamic processes in terms of (loop free) causal relationships among a set of internal variables
- relate this description to external variables that are considered to be manifestations of the internal processes
- describe various classifications imposed on the dynamic processes

CASNET models can be used to describe pathophysiological processes of disease. Knowledge is represented by three types of data elements corresponding to the three kinds of description outlined above: observations of the patient, pathophysiological

states and diagnostic prognostic and therapeutic categories. Observations are the direct evidence obtained about a patient. Pathophysiological states are intermediate constructs that describe internal conditions assumed to take place in the patient. They summarize results from many different observations. Categories of disease are conceptually at the highest level of abstraction summarizing patterns of state and observations.

When diagnosis is to be modelled in a domain of knowledge where mechanisms of disease are understood, the cause and effect model can be used to significantly improve the basis on which decisions are made. However, when less information is available, associations between findings must be relied on to a greater extent, and the goals of reaching structured and well explained conclusions and recommendations may not be fully satisfied.

2.3.5 The Hanna and Gold System

The Hanna and Gold [Hanna88] system is an expert diagnosis system employing both deep and shallow knowledge. The system under diagnosis may be dynamic; its components may have internal states and have varying propagation times; state transitions may be asynchronous, and feedback loops are allowed at all structural levels.

The system consists of two layers: a shallow expert (SE) which contains traditional symptom to fault associations along with a

hypothesis testing mechanism and a deep expert (DE) in which a hierarchical model of the diagnosed system represented in multiple levels of structural and behavioural abstraction

A multilevel simulator that shifts on demand from coarse qualitative modelling to detailed quantitative modelling assists the diagnosis process in verification and elimination of hypothesized suspects. Knowledge of pathological behaviour (failure nodes) of lower level components is incorporated in the simulator. Learning is exhibited as deep to shallow expertise transfer as well as up the abstraction levels of the simulator itself.

Integration between the DE and SE layers is manifested in two ways. (1) The diagnosis process recursively navigates through the levels of the system's structural hierarchy and at each level switches to deep simulation based expertise when the current shallow expertise is insufficient for solving the particular level. (2) As the deep expertise is being utilized now shallow expertise (i.e. direct symptom to fault assertion) is automatically constructed augmenting the SE layer for future more efficient diagnosis. The representation language and the inference engines of the two layers are implemented in PROLOG.

2.4 Where Present Work Fits In

How far in recent years research has been going on diagnostic systems in which the model incorporates some forms of incompleteness. While in the experiential approach as described above some attempts have been made to deal with incomplete and imperfect theories [Rajamoney87] the completeness of the model is a common assumption in first principle diagnostic systems. The assumption holds good in certain problem domains such as electronic troubleshooting and solutions from first principles have been successful. However this assumption is inadequate for the application of ontological approaches to complex problems such as medical diagnosis or mechanical troubleshooting. Indeed working at multiple levels of abstraction can reduce the computational complexity of the reasoning process. But then one still cannot assume that the lowest level of description is complete. Abstractions at the lowest level of domain theory imply missing knowledge. A further limitation of ontological approaches directly concerns the definition of diagnosis as a set of faulty components. The definition can be on the one hand too restrictive since we may want to distinguish between different faults of the same component on the other hand in many cases the faulty behaviour is not due to a small set of responsible faulty components.

The present work aims at providing a diagnostic mechanism based on a dual model of the domain which simultaneously incorporates some forms of incompleteness. The basic idea is in line with [Console89] (and also [Console88]). Summarization of shallow knowledge from the underlying deep model is also involved.

The main goals addressed are

- introducing a formalism to represent incomplete causal knowledge (elaborated in next chapter)
- giving a logical foundation of causal reasoning on incomplete knowledge
- providing a formal treatment of multiple fault diagnosis within causal models
- providing criteria to select most desirable solution of a problem
- abstraction of some shallow knowledge after finding a solution from the deep model

Chapter 3

SYSTEM LEVEL OVERVIEW

In this chapter an overview of the architecture of the diagnostic system developed is presented. First the theoretical basis is developed. Then a system level description is provided.

There seem to be two quite different approaches in the theory and design of diagnostic reasoning systems.

In the first approach, often referred to as diagnosis from first principles, one begins with a description of some system together with an observation of the system's behaviour. If this observation conflicts with the way the system is meant to behave, one is confronted with a diagnostic problem, namely to determine those system components which, when assumed to be functioning abnormally, will explain the discrepancy between the observed and the correct behaviour. For solving this diagnostic problem from first principles, the only available information is the system description, i.e. its design structure plus the observation(s) of the system behaviour. In particular, no heuristic information about system failures is available. Notable examples of approaches to diagnostic reasoning from first principles are ([Reiter87], [Davis84], [DeKleer85], [Reggia83], etc.). We shall call such approaches as ontological ones.

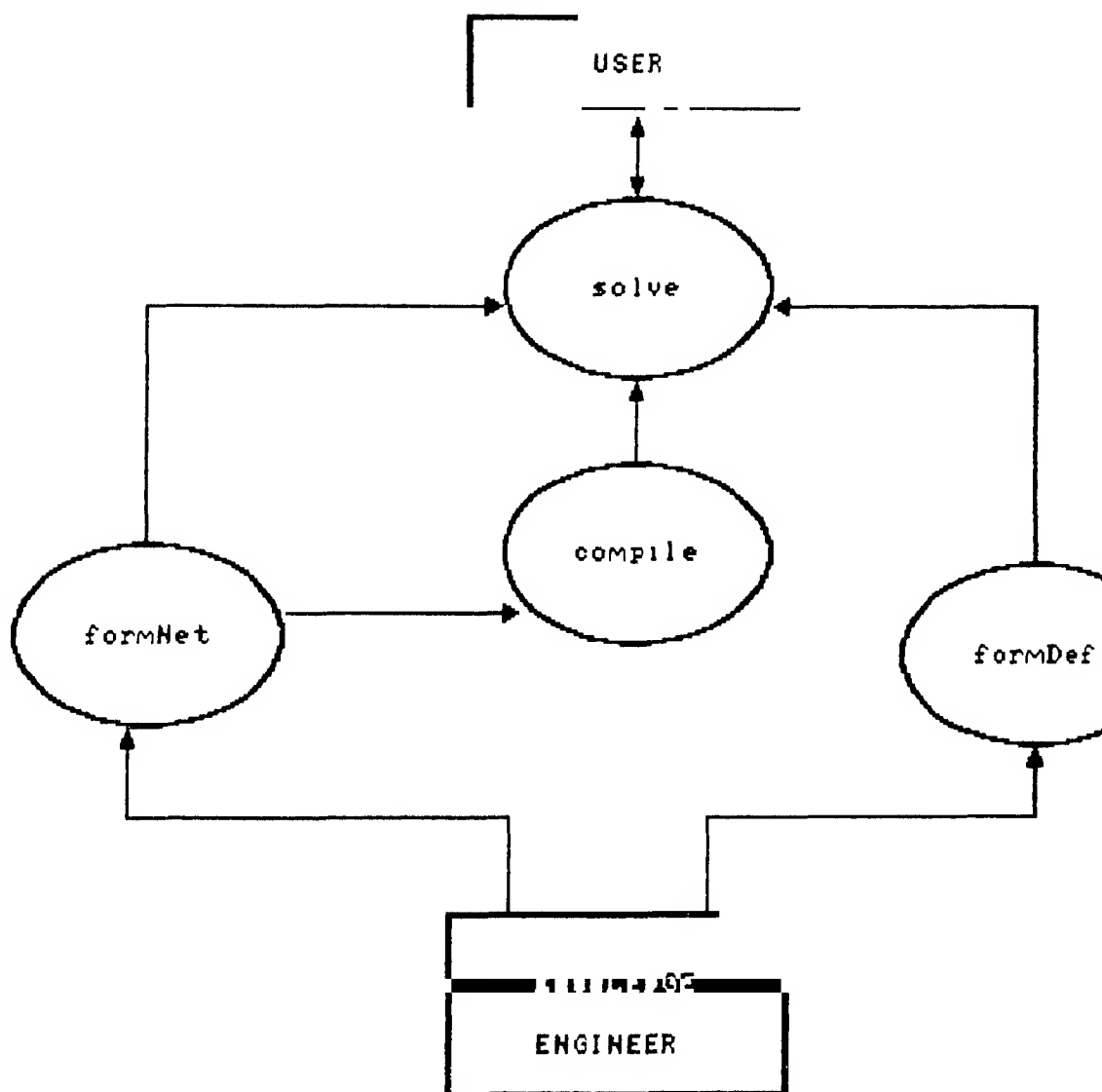


FIGURE 3 A SCHEMATIC DIAGRAM OF THE VARIOUS MODUL

Under the second approach to diagnostic reasoning which might be described as the experiential approach or diagnosis from experience heuristic information plays a dominant role. The corresponding diagnostic reasoning systems attempt to codify rules of thumb statistical intuitions and past experience.

However in recent years research has been going on diagnostic systems in which the model incorporate some form of incompleteness. This is only natural since many domains are so complex that some form of incompleteness is necessarily engrained in them and any diagnosis based solely on a so called complete model is bound to prove inadequate. While in the experiential approach some attempts have been made to deal with incomplete and imperfect theories [Rajamoney87] the completeness of the model is a common assumption in first principles diagnostic systems. The assumption holds good in certain problem domains such as electronic troubleshooting and solutions from first principles have been successful. However the assumption is inadequate for the application of ontological approaches to complex problems such as medical diagnosis or mechanical troubleshooting. Indeed working at multiple levels of abstraction can reduce the computational complexity of the reasoning process. But then one still cannot assume that the lowest level of description is complete. Abstractions at the lowest level of domain theory imply missing knowledge. A further limitation of ontological approaches directly concerns

th definition of diagnosis as a s t of faulty components The definition can be o the one hand too r strictiv nce we may want to distiguish between different faults of the same component on the other hand in many ca es the faulty behaviour is not due to a small et of responsible faulty components

Of late so called causal modelling approaches have b en widely adopted in diagnosis especially in m dical domains ([Clancey84] [Fink85]) Such approaches have widely b en applied to many eal world problems but they seem to b less constrained and well defined than ontological ones Some form of incompleteness need to be accommodated in causal models

The present work aims to provide a diagnostic mechanism i a causal modelling formalism with a precise logical formalization but which simultaneously incorporates some forms of incompleteness The basic idea is in lines with [Console89] (and al o [Console88])

3 1 A Formalism to Represent Deep Causal Knowledge

Ca sal network are a general formalism to rep es nt causal knowledge This knowledge may be used to describe the behaviours of a physical or physiological system We discuss how such causal networks can be us d to model the faulty behaviour of a system See the example in Figure 2 There are at least two advantages

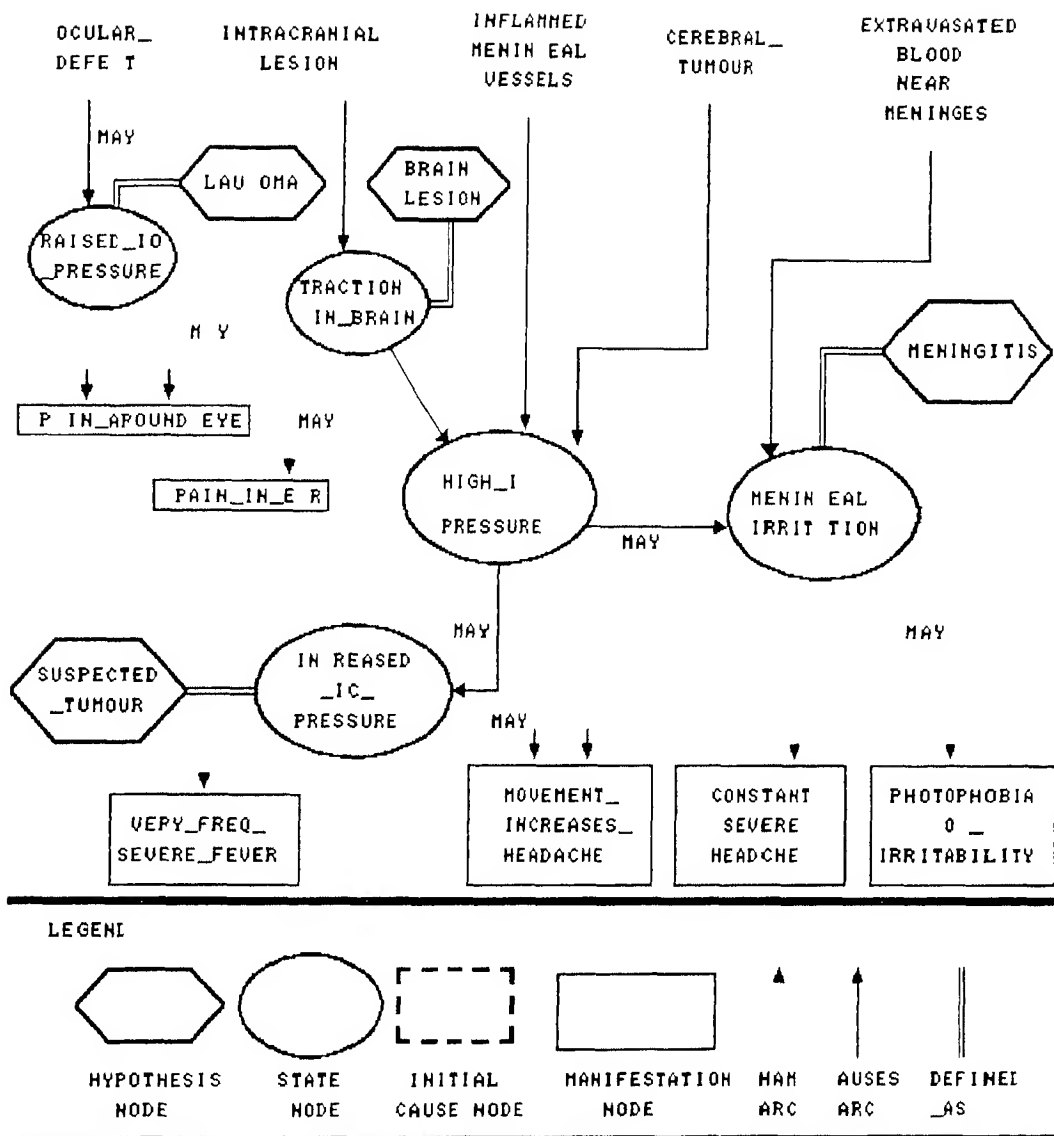


FIGURE 2 AN EXAMPLE CAUSAL NETWORK FROM THE MEDICAL DOMAIN

to this causal net representation. Hence, First, the system can trace from activated hypotheses backward along cause effect pathways to identify starting nodes in the network. Starting nodes are hypotheses for which no causes have been defined and are thus primary disorders. Second, if any intermediate node along this path is known to be false, this causal pathway can be ruled out as a candidate explanation for the patient's problem observations.

Four types of nodes are used in this causal modelling formalism. They are:

- (i) STATE nodes (elliptic boxes) representing partial states of the modelled system (these states are not observable).
- (ii) MANIFESTATION nodes (rectangular boxes) representing observable manifestations of internal states.
- (iii) INITIAL_CAUSE nodes (broken lined rectangular boxes) representing initial perturbations, the mechanisms starting the processes which might lead the system to a faulty behaviour. These are assumed to be non observable, introducing the first form of incompleteness in the model. Such nodes are abstractions of the actual perturbation processes and the model gives no direct way to establish their presence.
- (iv) HYPOTHESIS nodes (heagonal boxes) representing diagnostic hypotheses.

The nodes in a network can be connected by means of different types of arcs (relationships). They are

- (i) CAUSES arc (continuous arrow) connecting one or a set (conjunction) of INITIAL_CAUSE and STATE nodes to a STATE node. These arcs represent a cause-effect relationship.
- (ii) HAM arc (dashed arrow) connecting a STATE node to a MANIFESTATION node. These arcs connect states to their external manifestations. HAM stands as an abbreviation of Has As a Manifestation.
- (iii) DEFINED_AS arcs (double lines) connecting one or a set (conjunction) of STATE nodes to a HYPOTHESIS node. These represent that the hypothesis is defined as the presence of the states.

Hence hypotheses are defined concepts.

In case of different arcs entering a node N, such arcs are implicitly ORed. Besides the incompleteness associated with INITIAL_CAUSE nodes, some other form of incompleteness must be accommodated by causal modelling formalisms. Causal model of complex physical or physiological systems may be incomplete at least for the following reasons:

Choosing the level of detail of a model is domain and task dependent and often subjective. However, in order to constrain the complexity of a model, some bounds have to be

imposed to the level of detail at which knowledge is represented. Hence processes and conditions at a lower level than the chosen one have to be abstracted.

There exist (especially in physiological systems) processes and conditions which are not perfectly known.

To deal with such forms of incompleteness, the CAUSES and HAM relations can be labelled as MAY. A MAY label denotes that the model of the relationship is incomplete. That is, some condition or process has been abstracted. A relationship not explicitly labelled a MAY is assumed to be implicitly labelled MUST, i.e. fully specified.

3.2 Diagnostic Problems and their Solutions

Given the representation as above, a few definitions are now in place which are to be used later on.

Definition [D1]

The set of assumptions is the union of the sets of abstracted conditions and the initial causes.

Definition [D2]

Given a diagnostic hypothesis H , the definition of H denoted by $\text{def}(H)$ is the set $\{S_1, \dots, S_k\}$ of STATE node S_1, \dots, S_k which are connected to H by DEFINED_AS arcs.

Definition [D3]

A diagnostic problem $\langle d, p \rangle$ is a triple $\langle \text{NET}, \text{HYP}, \text{OBS} \rangle$ where

- NET is a set of nodes and arcs modelling the causal network NETWORK
- HYP = { $\langle H, \text{def}(H) \rangle$ | H is a HYPOTHESIS node in the causal network NETWORK }
- OBS is a nonempty set of ground manifestations i.e. observations. Thus OBS is always a subset of MANIFSET the set of all MANIFESTATION nodes

Since manifestations represent observable conditions we assume that manifestations not declared as observed are absent. The set of manifestations not observed is thus given by

$$\text{ABSENT} = \text{MANIFSET} - \text{OBS}$$

This assumption introduces non monotonicity in the diagnostic process in order to draw conclusions from incomplete knowledge. That is we only have a partial description of the case under examination.

A hypothetical world may be regarded as a tentative reconstruction of the causal evolution that has led to the observed situation. The notion of hypothetical world is defined recursively. The basis is the case of a world containing initial causes: the recursive case involves abstracted conditions on causal relationships.

Definition [D4]

Given a d p $P \langle \text{NET HYP OBS} \rangle$ and given a world W W a world for P iff

$W \in \text{NET} \cup \{c_1, c_2, \dots, c_k\}$ where c_i for $i = 1, \dots, k$ is a ground initial case (in this case we say that W is an initial world for P) or

$W = W_1 \cup \{A\}$ where

W_1 is a world for P

$X \& A \rightarrow Y$ is an instance of a formula in NET where A is an abstracted condition symbol $W_1 \models X$

In the latter case of the definition the world $W_1 \cup \{A\}$ has to be considered in order to examine the consequences of assuming that the condition A is actually true in the case under examination that is assuming that the corresponding MAY condition in the CAUSES or HAM relationship holds in the case under observation

Definition [D5]

Given a diagnostic problem $P \langle \text{NET HYP OBS} \rangle$ a world W for P is inconsistent iff $W \models \neg m$ for some ground manifestation m such that m is not an element of OBS

That is W is inconsistent for P iff W does not derive any manifestation which has not been observed and hence logically inconsistent

Definition [D6]

Given a world W and a set OBS of observations W covers OBS iff for all element m in OBS $W \models m$

Definition [D7]

Given a diagnostic problem $P = \langle NET \ HYP \ OBS \rangle$ a world W for P is a final world iff W covers OBS and is not inconsistent that is

$$OBS = \{ m / m \text{ is an observation } W \models m \}$$

Thus a final world W for P has to explain all the observed findings and must not predict the presence of manifestations that have not been found

Definition [D8]

Given a diagnostic problem $P = \langle NET \ HYP \ OBS \rangle$ and a final world W for P a solution to P is given by the set

$$\text{diagnosis}(W) = \{ H / \langle H \ \text{def}(H) \rangle \text{ is an element of } HYP \\ W \models \text{def}(H) \}$$

Then W is said to be a causal explanation of the solution

3.3 Parsimonious Solution

There may be more than one final world and then more than one solution to a diagnostic problem according to the definitions given above. A solution is one of the possible explanation of the given data. Thus we need to have some criterion with which we can say that one solution is better or more acceptable than

another. Then we can have a diagnostic algorithm which would allow us to have a solution which is better than all other solutions in the yardstick of the criterion.

The criterion chosen is that of parsimony. We would like to choose that solution which would involve the least number of assumptions over the conditions. The problem is that we have no direct way to establish the truth of the conditions that have been assumed in order to explain the observed findings. The assumptions made are on INITIAL_CAUSE nodes or on MAY arcs. Let us recall what the INITIAL_CAUSE nodes signify. They are abstractions of actual perturbation processes and the model gives no direct way to establish their presence. Now in many cases it is quite reasonable to assume that these perturbations are independent of each other. We too would derive solutions based on this assumption.

Each of these INITIAL_CAUSE nodes potentially gives rise to several MAY arcs which in turn form links in the paths to manifestations that have been observed. It is then more desirable that we select those INITIAL_CAUSE nodes earlier which potentially explain more of the observations. This is because of the assumption of independence we made above. The solution arrived at in this manner would then be parsimonious over INITIAL_CAUSE assumptions. We call such a solution an IC Parsimonious solution.

3.4 Problem Formulation & Generation of IC Parsimonious Solution

The system can be used to formulate a diagnostic problem and generate an IC Parsimonious solution. The following modules show how this is done. A schematic diagram is provided in Figure 3.

Module 1 Formation of the network

This module interactively lets the user form the network of the diagnostic problem domain. The network is gradually formed by asking the user to create a node and a link if further nodes are to be input.

Module 2 Defining the hypotheses

The definition of the hypotheses in terms of the various STATE nodes can be specified in this module. These model the DEFINED_AS arcs of the network. These relationships are at a different knowledge level than the other ones in the network and so are represented in a different manner.

Module 3 Compiling the network

This module compiles the network to form various lists. The knowledge acquired so far in the two previous modules is of a static nature in the sense that unless the underlying structure of the network changes, the acquired knowledge does not change. So it is advantageous to have compact chunks of knowledge that are derivable from the existing knowledge. This eliminates the need to unnecessarily repeat this process each time observations are acquired.

Module 4 Entering the observations

The manifestations which have been observed are input to this module. This along with the first two modules complete the formulation of the diagnostic problem.

Module 5 Generating the IC Parsimonious solution

This module uses the knowledge compiled in Module 3 and generates the IC Parsimonious solution. The solution is achieved in iterations of set covers over the set of all observations. The solution obtained is added as a shallow rule to the knowledge base so that when a solution is asked for the same set of observations the system consults the shallow knowledge to provide the solution.

Note that as the set of observations change the IC Parsimonious solution also changes. In particular when more observations enter the set OBSV some of the hypotheses confirmed earlier may no longer be predicted. This is consistent with the monotonicity that was introduced with the assumption that manifestations which are not declared as observed are absent.

3.5 An Example

Consider the following example from the medical domain. The causal model tries to capture some of the knowledge required to

diagnose ailments whose symptoms occur as various forms of headaches. Please refer to Figure 2.

The initial causes are ocular defect, intracranial lesion, inflamed meningeal vessels, cerebral tumour, and extravasated blood near meninges. These are initial perturbations to the system and according to our assumption not verifiable. The observable manifestations are pain around eye, pain in ear, very frequent and severe headache, movement increases headache, constant severe headache, and photophobia or irritability. The hypotheses are glaucoma (defined by the state raised_intraocular_pressure), suspected tumour (defined by the state increased_intracranial_pressure), brain lesion (defined by the state traction_in_brain), and meningitis (defined by the state meningeal_irritation). The different causal arcs which are labelled may are the may arcs. Those not labelled are the must arcs.

Chapter 4

Implementation Details

In this chapter the implementation details of the various system modules of the diagnostic system are presented. The system is developed on HP 9000 series in IF/Prolog ([Clocksin84]).

Sample sessions are provided in the appendices. Before we come to a module wise description of the system let us briefly touch upon the data structures used.

4.1 Data Structures

The two main data structures used are frames and lists. The network has been implemented in terms of frames. Each frame has the following optional slots:

- isa to describe the type of node
- causes to represent the CAUSES arc
- ham to represent the HAM arc

Each slot has an attribute and a value. The attribute in case of a isa slot is optional and in case of causes and ham slots can be must or may. The value in case of isa slot is the type of the node. In case of causes slot it is a STATE node while in case of a ham slot it is a MANIFESTATION node.

Lists have been used extensively to hold both local and global data structures. Prolog clauses have been used to create data bases of facts.

4 2 Implementation of the Various Modules

The implementation details of the various modules follow

4 2 1 Module formNet

This is the module which interactively lets the user form the network of the diagnostic problem domain. The network is gradually formed by asking the user to create a node and asking if further nodes are to be input. While creating a node the user is successively asked to input the slots. Proper validation is done while accepting the user input for the slot attribute and value. However, it is the onus of the knowledge engineer to create proper links between the network nodes and assign proper types for the nodes. This is because it is not possible to know beforehand the characteristics of the network in an arbitrary problem domain.

The following are the main routines used in this module

1 form_network

Form the network interactively

2 enquire_more_nodes(Ans)

Ask user if network formation is over and act depending on answer Ans

3 form_a_node

Form one node of the network

4 enter_slots(Node)

Enter the slots for this node Node

5 enter_a_slot(Node)

 Enter one slot of the node Node

6 enquire_more_slots Node Ans)

 Ask user if all slots for the node Node has been entered and
 act depending on answer Ans

7 check_slot(Slot)

 Validate the slot Slot

8 check_attrib(Slot Attrib)

 Validate the attribute Attribute for slot Slot

9 check_value(Slot Value)

 Validate the value Value for slot Slot

4 2 2 Module formDef

This module lets the knowledge engineer form the definitions of the hypothesis. Each hypothesis is defined as a conjunction of several STATE nodes. The final aim of the diagnosis is to generate a set of hypotheses as solution. While asking for definitions of the various hypotheses, the system validates input in the context of the network already formed.

The following are the main routines used in this module

1 form_all_definitions

 Ask user to input the hypotheses and form their definitions

2 form_a_definition(H StateSet)

 Form the definition for hypothesis H and validating against
 StateSet

3 get_definition_states(H Stateset)

 Get the states that define the hypothesis H validating against
 StateSet

4 `enquire_more_states(H StateSet)`

Enquire if more states are needed to define the hypothesis H

5 `check_state(S StateSet)`

Validate state S against StateSet

4.2.3 Module compile

This module compiles the knowledge entered so far in the modules `formNet` and `formDef` into chunks of knowledge to be later used in the diagnostic process. This part of the work is common across all diagnostic problems in where the causal network of the problem remains unchanged. For all such cases compiling the network is a one time effort. The actual diagnostic process can use the knowledge compiled to save a lot of repetitive work. In particular this module forms two lists corresponding to each `INITIAL_CAUSE` node

`MustList` this list contains a list of paths from the `INITIAL_CAUSE` node to the reachable `MANIFESTATION` nodes that contain no `MAY` arcs

`MayList` this list contains a list of paths from the `INITIAL_CAUSE` node to the reachable `MANIFESTATION` nodes each of which contains at least one `MAY` arc

The following is the main routine used in this module

1 `formMMList(InitCause Manif MustList MayList)`

Form the Must May lists given an initial cause `InitCause` and manifestation `Manif`

4 2 4 Module getObs

This modul allows the user to get the set of manifestations that ha e been observed. The manifestations not observed are assumed to be absent. This completes the specification of the diagnostic problem. This the part of the diagnostic problem liable to change more frequently than other parts.

Th following are the main routines used in this module

1 get_ob_ervations

Get the observations

2 enquire_manifestations(L)

Enquire if th manifestations in the list L have been observed

3 enquire_a_manif(M)

Enquire if the manifestation M has been observed

4 2 5 Module solve

This is the main module which actually gets the solution. Note that the solution cons sts of a set of hypothesis confirmed and hence involves multiple fault diag osis. Moreover the diagnosis is non monotonic. It makes use of the knowledge compiled in the module compile.

Th module uses the following algorithm

1 Partition the set of all INITIAL_CAUSEs into two sets VALID and INVALID wh re

VALID { IC | IC is an INITIAL_CAUSE node
 MustLeaves(IC) is a subset of OBSV }
 INVALID OBSV VALID
 OBSV set of all MANIFESTATIONs observed
 MustLeaves(IC) set of the last nodes in the
 elements of MustList(IC)
 MustList(IC) paths from the INITIAL_CAUSE node
 to the reachable MANIFESTATION nodes that contain
 no MAY arcs

2 Form the set VM where

VM union of MustLeaves(IC) over all IC where IC
 is an element of VALID

3 Form the set DIFF OBSV VM

/* If DIFF is a null set then solution is unique else there
 can be several ways to cover DIFF by forming unions over
 MANIFESTATION nodes that may be reached from the
 INITIAL_CONDITION nodes We look for the cover which is
 most parsimonious in the INITIAL_CONDITION nodes */

4 Let

INIT < VALID
 ADDITIONAL < NULL

5 Let

MayLeaves(IC) set of the last nodes in the elements of
 MayList(IC)

where

MayList(IC) paths from the INITIAL_CAUSE node to the
reachable MANIFESTATION nodes that
contain at least one MAY arc

6

/* find the set ADDITIONAL which would be containing
INITIAL_CAUSE nodes required in addition to those in
VALID to get a solution

Those INITIAL_CAUSE nodes are chosen earlier which may
lead to larger sets of MANIFESTATION node that have been
observed */

Repeat until DIFF = NULL

begin

for all IC in INIT do

form the set MayDiff(IC)

where

MayDiff(IC) intersection of the sets
MayLeaves(IC) and DIFF

Find ICK belonging to INIT such that

MayDiff(ICK) has the maximum cardinality among all

MayDiff(IC)s for ICs in INIT

ADDITIONAL ← union of ADDITIONAL and { ICK }

INIT ← delete ICK from INIT

DIFF ← DIFF MayDiff(ICK)

if INIT is null and DIFF is not null then

INIT < INVALID

end

7 Find the set ImposNodes where

ImposNodes set of all nodes that can never be included
in any possible cover

/* These are the nodes that must lead to observations that
have not been observed and hence according to our
assumption absent */

8 ValAd < union of VALID and ADDITIONAL

Cover d < null

9 For all IC in ValAd do

begin

put all nodes in all paths from IC to MustLeaves(IC)
which are not already in Covered in the set Covered

let MayObs(IC) intersection of the sets OBS and
MayLeaves(IC)

put all nodes in all paths from IC to MayObs(IC) which
are not already in Covered in the set Covered

end

/* the set Covered now contains all nodes in the network
which has been covered in forming the solution and form a
final world for the diagnostic problem (Definition D7)*/

```

10 Let HypoSet      set of all HYPOTHESIS node
      Def(H)        set of all STATE nodes that define the
                     HYPOTHESIS node H
      Solution <    null

```

```

11 For each element H in HypoSet do
      if D f(H) is a subset of Covered then put H in Solution

```

```

1  Output the set Solution which contains the IC Parsimonious
   solution

```

```

13 Add a shallow rule to the knowledge base abstracting the
    connection between the observations and the solution

```

While the solution is being formed a trail of the hypotheses entering the set Solution is displayed

The following are the main routines used in this module

```

1  get_path(Start Dest Route)
    Get the path from Start to Dest in Route
2  check_hypotheses(S)
    Form the IC Parsimonious hypothesis set S
3  cover_all_paths(L)
    Cover all paths starting from elements of the list L
4  assert_all(IC L)
    Assert all paths starting from IC to elements of the list
    passed as the second parameter

```

- 5 `formHypoSet(HypoSet)`
Form the set of all hypothesis
- 6 `cover_a_path(IC Obsv)`
Cover a path from IC to Obsv
- 7 `m x_cardinality(IL Diff ICmax MDmax Cmax)`
Find the initial condition ICmax having max cardinality of set MD among all ICs in IL the MD set corresponding to ICmax is returned in MDmax and the cardinality of the set MDmax is returned in Cmax
- 8 `formAdditional(ValidSet VMust Init Addl)`
The set Addl will contain the additional ICs required to cover all the observations
- 9 `formMD(IC MD Diff)`
Form the set MD
- 10 `formDiff(VMust Diff)`
Form the set Diff
- 11 `assertValidInvalid(ValidSet InvalidSet VMust)`
Form the sets Valid Invalid and VMust
- 12 `formICmustSet(IC ICmustSet)`
- 13 `formICmaySet(IC ICmaySet)`
Form the Must May lists for a given initial cause IC and a set of manifestations the lists contain the whole paths from the IC to manifestations
- 14 `formICmustLeaves(IC ICmustLeaves)`
ICmustLeaves contains only the manifestations that must be reachable from an initial condition IC

15 formICmayLeaves(IC ICmayLeaves)

ICmayLeaves contains only the manifestations that must be
reachable from an initial condition IC

16 form_one_ic_list(IC [] MustLists MayLists)

Form the MustList and MayList for one initial_cause

17 formObsvSet(ObsvSet)

Form the set of all observations

18 formManifSet(ManifSet)

Form the set of all manifestations

19 formImposLeaves(Impl)

Form the set of all impossible leaves the manifestations
that have not been observed

0 formImposNodes(ImpN)

1 assertImposNodes(ImpN)

Form and assert the set of all impossible non_leaf nodes
these nodes must lead to manifestations that have not been
observed

Chapter 5

CONCLUSIONS

5.1 Conclusions

The present work aims at providing a diagnostic mechanism based on a deep causal model which simultaneously incorporates some forms of incompleteness

The main goals addressed are

- introducing a formalism to represent incomplete causal knowledge

- giving a logical foundation of causal reasoning on incomplete knowledge

- providing a formal treatment of multiple fault diagnosis within causal models

- providing criteria to select most desirable solution of a problem

- abstraction of some shallow knowledge after finding a solution from the deep model

Traditional diagnostic systems have followed either the approach of reasoning from first principles with hardly any incompleteness in the model or the approach of reasoning guided by heuristics which lack in formulation of a deep model of the system under diagnosis. This work tries to explore diagnostic reasoning from

Chapter 5

CONCLUSIONS

5.1 Conclusions

The present work, aims at providing a diagnostic mechanism based on a deep causal model , which simultaneously incorporates some forms of incompleteness.

The main goals addressed are :

- introducing a formalism to represent incomplete causal knowledge ;
- giving a logical foundation of causal reasoning on incomplete knowledge ;
- providing a formal treatment of multiple-fault diagnosis within causal models;
- providing criteria to select most 'desirable' solution of a problem;
- abstraction of some shallow knowledge after finding a solution from the deep model;

Traditional diagnostic systems have followed either the approach of reasoning from first principles with hardly any incompleteness in the model or the approach of reasoning guided by heuristics which lack in formulation of a deep model of the system under diagnosis. This work tries to explore diagnostic reasoning from

first principles in a deep causal model of the system that incorporates two forms of incompleteness in the model. The first form of incompleteness is the assumption that initial perturbations that result in malfunctioning of the system are observable. The other incompleteness is the assumption that certain causalities are not precisely known. That is, given a state for the system one may give rise to the other but need not necessarily do so. This only implies that certain conditions are being abstracted. Such abstraction is sometimes done in order to constrain the complexity of the model. On the other hand, where the processes and conditions are not perfectly known (for example in physiological systems) such abstraction cannot be avoided.

The diagnostic process as described in the present work is non-monotonic in the sense that a solution for a problem need not be a subset of a solution of a problem whose set of observations is a superset of the observations of the former. Also since the model incorporates incomplete knowledge there may be multiple solutions to the same problem. (However each solution may consist of multiple faults.) Now among all such solutions we need to find out the one which is best in some sense. The criterion of parsimony is often used to rank solutions. In this case we have chosen parsimony based on initial causes and the solution obtained is termed IC Parsimonious solution. The reason why the

criterion is chosen, is that the initial causes are not observable according to our assumption. Each such initial cause potentially gives rise to manifestations which must be observed or which may be observed. The parsimony is enforced by first including the initial causes whose "must" manifestations (that is those manifestations which must be observed if that initial cause is actually present) have been observed, followed by including the initial causes with observed "may" manifestations (i.e., those manifestations which may be observed if that initial cause is actually present) in the set of nodes covered.

The other aspect is the realisation that it is not always that a diagnosis is based on a deep model of the system. One tries to reason on a shallow model as long as one can afford, and fall back on the deep model when the shallow knowledge does not suffice. This means that for a problem not known before the diagnosis necessarily goes on a deep model. But next time when the problem is encountered, the reasoning goes on a shallow model. So when a new problem is faced, apart from diagnosing on a deep model, some knowledge is abstracted and put in the shallow model of the system. The present work aims at doing this by abstracting shallow knowledge (rules) from a diagnosis based on deep, incomplete causal model of the system. Note, that the solution is the same for a given problem, whether deep or shallow reasoning is involved. In case shallow reasoning can provide a solution, it is as good or as bad as that of the deep model.

However reasoning based on a deep model can potentially provide deeper explanations than reasoning based on a shallow model

5.2 Improvements and Scope for Future Work

The diagnostic system developed can be improved in several ways. First and foremost good explanation facility can be provided as reasoning goes on. One can also explore multi-paradigm representations of both deep and shallow knowledge and also reasoning in which both deep and shallow knowledge cooperate. The other aspect is trying to quantify the incompleteness in the model in some way and generate some confidence measures associated with solutions. One can explore assigning confidence measures associated with the must and may arcs and combining these measures in some way as the reasoning goes on. In that case a parsimony criterion may be based on minimality of the combination of such measures. However assigning the measures and combining them suitably do remain open to subjective judgements.

REFERENCES

[Aye88]

Aye B El and Finance J P
On Cooperation Between Deep and Shallow Reasoning SIDI
An Expert System for Troubleshooting Diagnosis in Large
Industrial Plants
Artificial Intelligence in Engineering Diagnosis & Learning
Elsevier 1988

[Clancey84]

Clancey and Shortliffe
Readings in Medical Artificial Intelligence
Addison Wesley 1984

[Clocksin84]

Clocksin W and Mellish C
Programming in Prolog
Springer Verlag New York 1981

[Console88]

Console L Dupre T Dupre and Torasso P
A Logical Approach to Deal with Incomplete Causal Models in
Diagnostic Problem Solving
Lecture Notes in Computer Science 313 255 264
Springer Verlag 1988

[Console89]

Console L Dupre T Dupre and Torasso P
A Theory of Diagnosis for Incomplete Causal Models
International Joint Conference on Artificial Intelligence
1311 1317 1989

[Davis84]

Davis R
Diagnostic Behaviour Based on Structure and Behavior
Artificial Intelligence 24 1984

[DeKleer85]

De Kleer J and Williams B C
Diagnosing Multiple Possibly Intermittent Faults or
Troubleshooting without Backtracking
Technical Report
Xerox Palo Alto Research Center Palo Alto CA 1985

[DeKleer87]

De Kleer J and Williams B C
Diagnosing Multiple Faults
Artificial Intelligence 32 97 130 1987

[Fink87]

Fink P and Lusth C
Expert Systems and Diagnostic Expertise in the Mechanical
and Electrical Domains
IEEE Transactions on Systems Man and Cybernetics(SMC)
17(3) 1987

[Hanna88]

Abu Hanna A and Gold Y
An Integrated Deep shallow Expert System for Mult level
Diagnosis of Dynamic Systems
Artificial Intelligence in Engineering Diagnosis & Learning
Elsevier 1988

[Johnson88]

Johnson L and Keravnou E T
Expert System Architectures
Kogan Page 1988

[Moen88]

Johnson Paul E Moen James B and Thompson W B
Garden Path Errors in Diagnostic Reasoning
Expert System Applications
Bolc L and Coombs M J (Editors)
Springer Verlag 1988

[Rajamoney87]

Rajamoney S and Dejong G
The Classification Detection and Handling of Imp rfect
Theory Problems
International Jo nt Conference on Artificial Intelligence
IJCAI 87 (Milano 1987)

[Reggia83]

Reggia J A Nau D S and Wang Y
Diagn stic Expe t Systems Based on a Set Covering Model
International Journal of Man Machine Studies 19 1983

[Reiter87]

Reiter R

A Theory of Diagnosis from First Principles
Artificial Intelligence Vol 32 No 1 3 1987

[Saucier89]

Saucier G Ambler A and Breuer M A (Editors)
Knowledge Based Systems for Test and Diagnosis
North Holland 1989

[Wilkinson85]

Wilkinson A J

MIND an Inside Look at an Expert System for Electronic
Diagnosis
IEEE Design & Test of Computers 3(4) 1985

[Weiss78]

Weiss S M Kulikowski C A Amarel S and Safir A
A Model Based Method for Computer Aided Medical Decision
Making
Artificial Intelligence 11
North Holland Publishing Company 1978

APPENDIX A

Sample Network Formation

```
/*  
*****
```

Here is part of a sample session for forming a network

```
*****  
*/
```

? formnet

Enter the node name

Node ocular_defect

Enter the slots for this node

Slot Name isa

Attribute

Value initial

Permitted values are [initial_cause state manifestation]
Please enter again

Value initial_cause

More slots to enter ? [y/n]

y

Node ocular_defect

Slot Name causes

Attribute may

Value raised_io_pressure

More slots to enter ? [y/n]

n

More nodes to enter ? [y/]

y

Enter the node name

Node raised_io_press re

Enter the slots for this node

Slot Name isa

Attribute

Value stat

More slots to enter ? [y/n]

y

Node raised_io_pre sure

Slot Name causes

Attribute must

Value pain_around_eye

More slots to enter ? [y/n]

n

More nodes to enter ? [y/n]

y

Enter the node name

Node pain_around_ ye

Enter the slots for this node

Slot Name isa

Attribute

Value manifestation

More slots to enter ? [y/n]

n

More nodes to enter ? [y/n]

n

?

APPENDIX B

Sample Diagnostic Session

/*

Here is a session of diagnosis for the network shown
in figure 2

*/

/* load all relevant files */

? demoha

consult file nsha loaded in 1 sec

consult file formdef loaded in 0 sec

consult file getobs loaded in 0 sec

consult file hanet loaded in 1 sec

IF/Prolog Version 3 4 2 HP 9000/800 created 1/9/88

Copyright (C) 1984 88 InterFace Computer GmbH

/* form the definitions */

? fd

Enter the hypotheses

Hypothesis glaucoma

Any more hypothesis ? (y/n) y

Hypothesis brain_lesion

Any more hypothesis ? (y/n) y

Hypothesis suspected_tumour

Any more hypothesis ? (y/n) y

Hypothesis meningitis

Any more hypothesis ? (y/n) n

The list of hypotheses is

- - - - -

[glaucoma brain_lesion suspected_tumour meningitis]

Press <Return> to continue

Definition for the hypothesis glaucoma

Enter the states that define this hypothesis glaucoma

Enter State raised_io_pressure

More states for this definition of glaucoma ? n

Definition for the hypothesis brain_lesion

Enter the states that define this hypothesis brain_lesion

Enter State traction_in_brain

Definition for the hypothesis glaucoma

Enter the states that define this hypothesis glaucoma

Enter State raised_io_pressure

More states for this definition of glaucoma ? n

Definition for the hypothesis brain_lesion

Enter the states that define this hypothesis brain_lesion

Enter State traction_in_brain

More states for this definition of brain_lesion ? n

Definition for the hypothesis suspected_tumour

Enter the states that define this hypothesis suspected_tumour

Enter State increased_io_pressure

More states for this definition of suspected_tumour ? n

Definition for the hypothesis meningitis

Enter the states that define this hypothesis meningitis

Enter State : meningeal_irritation

More states for this definition of meningitis ? n

No more hypotheses !!

yes

/* compile the network to form different lists */

?- compile.

The lists for the initial condition : ocular_defect

=====

MustLists :

[]

Press <Return> to continue ...

MayLists :

[[ocular_defect,raised_io_pressure,pain_around_eye]]

Press <Return> to continue ...

The lists for the initial condition : intracranial_lesion

=====

MustLists :

[]

Press <Return> to continue ...

MayLists :

[[intracranial_lesion,traction_in_brain,high_ic_pressure,
meningeal_irritation,photophobia_or_irritability],
[intracranial_lesion,traction_in_brain,high_ic_pressure,
meningeal_irritation,constant_severe_headache],
[intracranial_lesion,traction_in_brain,high_ic_pressure,
meningeal_irritation,movement_increases_headache],
[intracranial_lesion,traction_in_brain,high_ic_pressure,
increased_ic_pressure,very_frequent_severe_headache],
[intracranial_lesion,traction_in_brain,pain_in_ear],
[intracranial_lesion,traction_in_brain,pain_around_eye]]

Press <Return> to continue

The lists for the initial condition inflamed_meningeal_vessels

MustLists

[]

Press <Return> to continue

MayLists

```
[[inflamed_meningeal_vessels high_ic_pressure
meningeal_irritation photophobia_or_irritability]
[inflamed_meningeal_vessels high_ic_pressure
meningeal_irritation constant_severe_headache]
[inflamed_meningeal_vessels high_ic_pressure
meningeal_irritation movement_increases_headache ]
[inflamed_meningeal_vessels high_ic_pressure
increased_ic_pressure very_frequent_severe_headache]]
```

Press <Return> to continue

The lists for the initial condition cerebral_tumour

MustLists

[]

Press <Return> to continue

MayLists

```
[[cerebral_tumour high_ic_pressure meningeal_irritation
photophobia_or_irritability]
[cerebral_tumour high_ic_pressure meningeal_irritation
constant_severe_headache]
[cerebral_tumour high_ic_pressure meningeal_irritation
movement_increases_headache ]
[cerebral_tumour high_ic_pressure increased_ic_pressure
very_frequent_severe_headache]]
```

Press <Return> to continue ...
The lists for the initial condition : extravasated_blood_near_meninges
=====

MustLists :

[[extravasated_blood_near_meninges,meningeal_irritation,
constant_severe_headache],
[extravasated_blood_near_meninges,meningeal_irritation,
movement_increases_headache]]

Press <Return> to continue ...

MayLists :

[[extravasated_blood_near_meninges,meningeal_irritation,
photophobia_or_irritability]]

Press <Return> to continue ...

yes

/* get the observations */

?- getobs.

The list of manifestations is :

[pain_around_eye,pain_in_ear,very_frequent_severe_headache,
movement_increases_headache,constant_severe_headache,
photophobia_or_irritability]

Which of these manifestations have been observed ?

Observed pain_around_eye ? [y/n] n

Observed pain_in_ear ? [y/n] n

Observed very_frequent_severe_headache ? [y/n] n

Observed movement_increases_headache ? [y/n] y

Observed constant_severe_headache ? [y/n] y

Observed photophobia_or_irritability ? [y/n] y

No more manifestations !!

yes

/* now solve the problem */

? solve

ValidSet [extravasated_blood_near_meninges]

InvalidSet [ocular_defect intracranial_lesion
inflamed_meningeal_vessels_cerebral_tumour]

VMust [constant_severe_headache_movement_increases_headache]

Impossible non-leaf nodes [increased_ic_pressure_raised_o_pressure]

Covered List [photophobia_or_irritability_movement_increases_headache
constant_severe_headache_meningeal_irritation
extravasated_blood_near_meninges_dummy]

Press <Return> to continue

The Trail of confirmed hypothesis so far [meningitis]

Press <Return> to continue

The IC Parsimonious hypothesis set [meningitis]

yes

/* next time if the same problem is solved shallow
knowledge is consulted to provide the solution
*/

? solve

The confirmed hypothesis set [meningitis]

yes

/* end session */

? by

112228

Date Slip

This book is to be returned on the
date last stamped

--

CSE-1991-M-BHA DIA